

ОСНОВНІ АСПЕКТИ ВИКОРИСТАННЯ WEB-ТЕХНОЛОГІЙ ПРИ РОЗРОБЦІ АРХІТЕКТУРИ МЕРЕЖЕВОГО РЕСУРСУ

Розглядаються типові архітектури інформаційних web-систем. Проаналізовано найбільш розповсюджені web-технології та їх використання в інформаційних системах. Акцентовано увагу на використанні безкоштовних web-технологій. Представлено архітектуру інформаційної web-системи та web-технології для діагностики та оцінки організаційної культури вищого навчального закладу.

Ключові слова: web-система, web-технології, архітектура, Client-Side, Server-Side, JavaScript.

Рассматриваются типичные архитектуры информационных web-систем. Проанализированы наиболее распространенные web-технологии и их использование в информационных системах. Акцентировано внимание на использовании бесплатных web-технологий. Представлена архитектура информационной web-системы и web-технологии для диагностики и оценки организационной культуры высшего учебного заведения.

Ключевые слова: web-система, web-технологии, архитектура, Client-Side, Server-Side, JavaScript.

The typical architecture of web-information systems is considered. The most common web-technologies and their use in information systems is analyzed. Using free web-technologies is paid attention. The architecture of information systems and web-technologies to diagnose and assess the organizational culture of higher education is represented.

Key words: web-system, web-technology, architecture, Client-Side, Server-Side, JavaScript.

Загальна постановка проблеми та її зв'язок з науково-практичними задачами. При створенні будь-якої системи потрібно приділяти увагу на етапі проектування вибору архітектури та технології які будуть використовуватись. Проблема вибору архітектури та web-технологій для інформаційної системи орієнтованої на web-доступ (Web-Based) для університету є актуальною, так як існує багато архітектур та технологій. Оптимальний вибір забезпечить розширений термін життєвого циклу системи, просту підтримку та зручність у використанні університетом у короткостроковій та довгостроковій перспективі.

Постановка задачі. Постає питання які технології та яка архітектура інформаційних web-систем для університету будуть відповідати таким умовам: простота у супроводженні, – користуванні, широка розповсюдженість (легко знайти технічних спеціалістів для супроводження, підтримки системи, модернізації, розширення тощо). Обрання систем орієнтованих на веб-доступ пояснюється досить просто – кросплатформеність, спрощена система використання «Logon and Go» – потрібен лише браузер, логін та пароль, доступ з корпоративної локальної мережі (Intranet) – будь-яке підключення; якщо потрібен доступ з мережі Internet – доступ до Internet і системою вже можна користуватись. Отже задачею даної статті є проаналізувати використання web-технологій в архітектурі інформаційних web-систем; обрати найбільш оптимальний варіант (комбінацію) для використання університетом. В результаті буде запропоновано архітектуру web-системи та технології які потрібно використати вищим навчальним закладом для задоволення потреб у навчально-педагогічній та науково-дослідницькій діяльності.

Огляд публікації та аналіз невирішених проблем. Сьогодні існує безліч публікацій на дану тематику [1, 2, 3], але як правило розглядаються окремо архітектура та технології. Проте система функціонує в цілому й потрібно розглядати на різних рівнях її функціонування та їх взаємодію. Така увага дозволить визначитись з вимогами як до програмної так апаратної частини інформаційної системи.

Основна частина. Головною складовою системи, що розроблюється є її архітектура. Від того наскільки детально вона пророблена залежить доля системи. Також вона повинна відповідати бізнес-

задачам організації. Головною особливістю некомерційних організацій є необхідність у відповідності умові «ціна-якість». Тобто присутня обмеженість ресурсів, що значно зменшує кількість варіантів архітектурних рішень. Звичайно й для комерційних організацій важливе те ж питання.

Розглянемо типові блоки (компоненти) архітектури інформаційних web-систем: клієнтська частина (Client-Side) – складається з браузера (веб-браузера) програми яка забезпечує відображення контенту, всі операції виконуються на стороні клієнта (які можна виконувати) [4]; серверна частина (Server-Side) – отримання запитів від client-side та їх обробка, повернення результатів клієнту, також включає набір серверів з відповідним програмним забезпеченням [5].

На рис. 1. представлено спрощену архітектуру інформаційної системи з веб-доступом. Як вже зазначалось вище є дві частини (сторони, sides). Каналами для передачі запитів від користувача до серверу є локальна мережа, як правило Internet, Intranet. При використанні першої присутнє досить суттєве обмеження пропускної спроможності. Для другої типовим є 100 Мбітні мережі, які зараз можуть замінюватись 1 Гігабітними, в залежності від потреб організації.

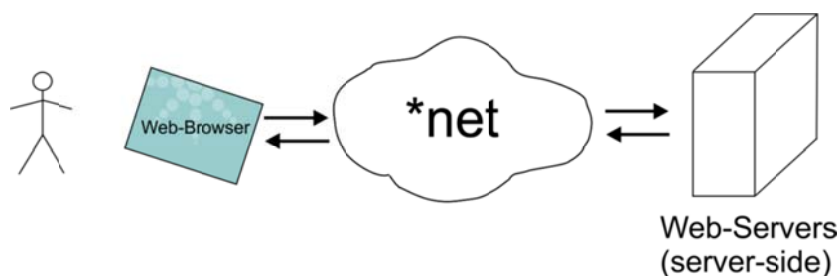


Рис. 1. Спрощена архітектура web-систем

Розглянемо більш детально Client-Side. Найбільш важливу роль відіграє браузер. Його роль полягає у забезпеченні зручного інтуїтивно зрозумілого інтерфейсу взаємодії з користувачем. На сьогодні існує багато версій браузерів. Головна перевага полягає в тому, що доступ до інформаційної web-системи можна отримати майже з будь-якого цифрового пристрою, який має браузер: комп'ютер, телефон, смартфон, планшетний комп'ютер тощо. Звичайно система повинна бути оптимізована під типові характеристики згаданих апаратних засобів, зокрема досить важливим є збереження зручного відображення інформації з використанням усієї області екрану.

Розповсюдженість браузерів потрібно враховувати при розробці front-end (те що бачить користувач через браузер) та Back-End (те що бачить адміністратор сайту, та використовує для наповнення контентом систему) інтерфейсу. У таблиці 1 представлено порівняння ринкових часток браузерів [6, 7, 8]. Проте є похибки, через те що деякі браузери, наприклад Opera, можуть видавати себе за інші.

Таблиця 1

Відсоткові частки (долі) браузерів (розповсюдженість у світі)

Рік/квартал	Internet Explorer	Firefox	Chrome	Safari	Opera
2011/I	50,19	24,90	12,35	6,17	2,13
2010/IV	52,01	25,60	10,76	5,70	2,09
Середнє значення	52,49	24,55	11,30	5,99	2,14

Найбільш розповсюдженим є Internet Explorer, менш – Opera. Варто зазначити, що також є й інші браузери, але їх доля не більше 1-1,5 % (у таблиці відсутні). Версії браузерів опущено. Всі значення розраховано як середнє арифметичне з даних на сайтах [6, 7, 8]. Тобто розроблюючи інформаційну web-систему потрібно орієнтуватись на роботу з першими чотирма браузерами, так як їх частки досить вагомі.

Порівнювати можливості браузерів не будемо, лише зазначимо, що є певні специфічні особливості обробки HTML (HyperText Markup Language – це мова для розмітки гіпертексту), JavaScript, CSS (Cascading Style Sheets – каскадні таблиці стилів, візуалізація веб-сторінок) у кожному з них, що потрібно враховувати при розробці інтерфейсу з яким взаємодіє користувач.

Для клієнтської частини (Front-End) використовується HTML+JavaScript+CSS. Також використовують Adobe Flash, Microsoft Silverlight – проте приділяти увагу даним засобам не будемо, так як їх використання значно збільшує апаратні вимоги до комп'ютеру користувача, а також потребує встановлення відповідних плагінів (plugin) та їх підтримка є обмеженою, або відсутня на деяких пристроях (смартфонах тощо).

Використання «чистого» JavaScript позбавлено сенсу, так як вже готові рішення. Тобто використання фреймворків (framework, фреймворк – система, набір готових засобів, що дозволяє працювати на більш високому рівні, тобто не звертати увагу на реалізацію потрібних методів та засобів) зменшує кількість часу на створення елементів інтерфейсу.

На даний час широкого розповсюдження набули різноманітні JavaScript фреймворки (framework): jQuery, MooTools, jQuery UI, Prototype, YUI, dojo, ExtJs (наведено від найбільш розповсюдженої до

найменш). Звичайно кожен має свої переваги та недоліки. Також й різні типи ліцензій: GPL, MIT, commercial. Наведеним вище умовам відповідає jQuery та jQuery UI (створена поверх основної бібліотеки jQuery), тип ліцензії MIT (розроблена Масачусетським технологічним інститутом, надає можливість змінювати в розроблених продуктах тип ліцензії, наприклад, для закриття коду та комерціалізації проекту; також варто зазначити, що дана ліцензія визнана придатною для використання в сфері наукових розробок).

Генерацією результуючого коду займається веб-сервер, на якому встановлено відповідне програмне забезпечення. Розглянемо з чого складається Server-Side. Перш за все Data Processing – обробка даних, виконання скриптів, кодів програм; Data Storage – місце де зберігаються дані, як правило розділяються на дані-коди програм, source коди, та бази даних (у фізичному змісті – файли). Найпростіший варіант Server-Side, коли веб-сервер (web-server – WS), сервер баз даних (DataBase Server – DBS), файловий сервер File Server консолідовано на одному фізичному сервері. Перевагами є невелика ціна, простота у підтримці. Недоліки – відсутність відмовостійкості, обмежена швидкодія. Такий варіант є найбільш оптимальним для університету, доки не потрібно підвищена відмовостійкість та більша швидкодія.

Найбільш простим способом розширення даного рішення є рознесення всіх згаданих вище логічних ролей серверів на окремі фізичні сервери.

Звичайно в даному варіанті підвищено відмовостійкість – вихід з ладу, наприклад веб-серверу не впливає на дані, але потрібно забезпечити резервне копіювання вже трьох серверів та більш швидку мережу для взаємодії між серверами.

Для забезпечення найбільшої відмовостійкості та швидкодії будуються кластери (High Availability cluster – HA cluster) з серверами для балансування навантажень (Load Balancers – LB). Звичайно можливо багато й інших варіантів архітектур, що як правило підбирається під конкретні умови та потреби організації.

Для аналізу взаємодії кожного компоненту в архітектурі інформаційних web-систем розглянемо приклад веб-орієнтованої системи для діагностики та оцінки організаційної культури університету. Дані система повинна забезпечити онлайн опитування експертів, зберігання/виведення даних та їх аналіз. Вимоги до системи: простота у підтримці та супроводженні, відповідність «ціна-якість». Потрібно обрати засоби та технології які будуть використовуватись в архітектурі розроблюваної web-системи. Головним елементом буде Система керування навчання (LMS). Вона дозволить поєднати наукові дослідження та забезпечення навчального процесу з використанням дистанційної освіти, тобто забезпечить науково-дослідницькі та навчально-педагогічні потреби університету. У якості LMS буде Moodle (модульне об'єктно-орієнтоване середовище дистанційного навчання – Modular Object-Oriented Dynamic Learning Environment), як широко розповсюджена [10] та більш функціональна порівняно з іншими безкоштовними LMS [11, 12]. Обрання Moodle також пов'язано з мовою на якій написано LMS – PHP. Є найбільш розповсюдженою скрипковою мовою для розробки web-систем. Більшість LMS розроблені з її використанням. Типовим набором є використання PHP + MySQL. Наявність безкоштовних open-source LMS на інших мовах: Java, ASP.NET, Python досить низька. Використання Java та ASP.NET суттєво збільшує вимоги до апаратного забезпечення сервера та потребує більш складного процесу налаштування. Для використання LMS розробленої з використання ASP.NET потребує ліцензію на операційну систему (ОС) Microsoft Windows Web Server 2008 R2, що збільшує витрати університету порівняно з використанням PHP + ОС Linux. Також суттєвою перевагою PHP є велика кількість спеціалістів та відносна простота освоєння мови. Front-End – для відображення, реалізації інтуїтивно зрозумілого інтерфейсу користувача – фреймворк jQuery. Обрання LMS Moodle

Стосовно Server-Side необхідно обрати ОС, як вище вже зазначалось, оптимальним вибором буде ОС з сімейства Linux. Використання безкоштовної ОС заощадить матеріальні кошти; відсутність графічного інтерфейсу (варто зазначити, що його можна встановити, але це лише поїдає цінні ресурси серверу та не спрощує підтримку, адміністрування, так як все одно потрібно використовувати командний рядок – shell) зменшить вимоги до апаратного забезпечення сервера та збільшить захищеність серверу, так як чим менше програмного забезпечення використовується тим менше шляхів для проникнення в ОС. Звичайно є ряд широко відомих дистрибутивів: Debian, CentOS, Ubuntu Server, Gentoo тощо, тому потрібно обрати один. Найбільш розповсюджені наведені вище.

Коротко наведемо переваги та недоліки кожного дистрибутиву. Debian – найбільш стабільний, висока якість, є базою для більше ніж 120 похідних дистрибутивів (наприклад, Ubuntu), підтримка багатьох процесорних архітектур, підтримка великої кількості пристроїв, швидкий випуск оновлень безпеки (security updates); але повільний випуск нових версій, відсутня підтримка деяких новітніх технологій, які як правило не довгий час на ринку. CentOS – стабільний, підтримка пакетної бази комерційного Red Hat Linux Enterprise; недоліки аналогічні Debian. Ubuntu Server – оснований на нестійкій гілці (пакети, які знаходяться у розробці) Debian, що є досить суттєвим недоліком при використанні у якості серверної ОС. Gentoo – дозволяє максимально оптимізувати ОС під сервер на який встановлюється, висока гнучкість налаштування системи; проте найбільш суттєвий недолік необхідність компіляції, внесення великої кількості налаштувань, як наслідок витрачання багато часу на етапі встановлення та складність підтримки (великий ризик зламати всю систему) [13, 14]. Також варто звернути уваги на те, що перший дистрибутив простіше в підтримці. Тому обираємо Debian. У подальшому якщо виникне необхідність забезпечити високу доступність веб-серверів, кластери, на Debian це досить простий процес, так як вже є

багато готових рішень, наприклад, High Availability (HA – кластер високої доступності) можна розгорнути менше, ніж за дві години використовуючи технічні статті та документації в мережі Інтернет.

Так як обрано LMS Moodle, яка підтримує MySQL та PostgreSQL, потрібно серед цих двох систем керування базами даних обрати одну, але варто зазначити якщо виникне ситуація, що можливостей MySQL не достатньо можна перейти на PostgreSQL. Суттєвою перевагою MySQL є простота в налаштуванні, підтримці та обслуговуванні, а при необхідності використання відмовостійких баз даних, потрібно використати реплікацію MySQL, яку можна налаштувати без досвіду, звичайно використовуючи технічну документація, менше ніж за три-чотири години.

На рис. 2. представлено остаточну архітектуру web-системи, на якій зображено усі компоненти, зокрема веб-технології:

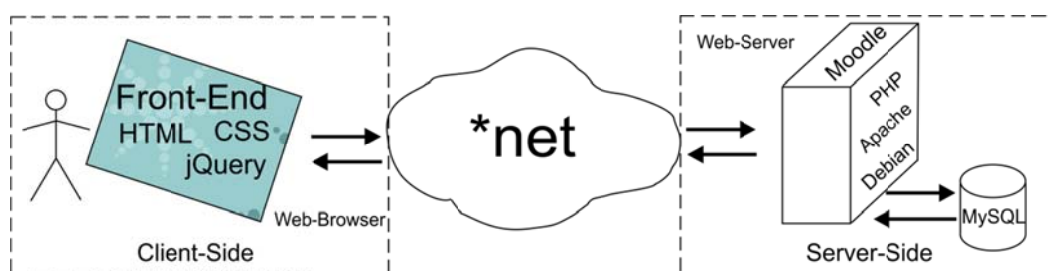


Рис. 2. Архітектура з обраними веб-технологіями

Висновки Таким чином типова архітектура складається з Client-Side та Server-Side. Використання безкоштовного програмного забезпечення дозволяє заощади. Усі наведені веб-технології є досить типовим набором у сфері інформаційних веб-систем, що нам надає гнучкість та простоту при обслуговуванні, так як рішення типових потреб та проблем вже існують у великій кількості, що спрощує процес супроводження системи, пошук технічних спеціалістів.

Отже обрано засоби для створення інформаційної веб-системи для діагностики та оцінки організаційної культури університету. Потрібно розробити інтуїтивно зрозумілий інтерфейс користувача, забезпечити найбільш просте відображення питань, даних та надати можливість діагностувати та оцінювати організаційну культуру університету на життєвих етапах організації. Варто акцентувати увагу на особисті якості менеджменту, як головну складову при зміні типу організаційної культури та досягнення стратегічних цілей організації.

ЛІТЕРАТУРА

1. Ільєнко Ю. І. Сучасні Web-технології [Електронний ресурс] / Ю. І. Ільєнко, Т. І. Кирилук – Режим доступу : URL : http://www.rusnauka.com/1_NIO_2011/Informatica/78189.doc.htm. – Загол. з екрану.
2. Крупин А. Архитектура информационных систем [Електронний ресурс]. – Режим доступу : URL : <http://old.cio-world.ru/infrastructure/29164/>. – Загол. з екрану.
3. Hodgdon J. Which Web Technology Does What? [Електронний ресурс] / Jennifer Hodgdon. – Режим доступу : URL : http://www.poplarware.com/articles/which_does_what. – Загол. з екрану.
4. Client-side [Електронний ресурс] // Wikipedia. – Режим доступу : URL : <http://en.wikipedia.org/wiki/Client-side>. – Загол. з екрану.
5. Server-side [Електронний ресурс] // Wikipedia. – Режим доступу : URL : <http://en.wikipedia.org/wiki/Server-side>. – Загол. з екрану.
6. Browser Statistics [Електронний ресурс]. – Режим доступу : URL : <http://marketshare.hitslink.com/browser-market-share.aspx?qprid=2&qptimeframe=Q&qpsp=48>. – Загол. з екрану.
7. Browser Version Market Share [Електронний ресурс]. – Режим доступу : URL : <http://www.w3counter.com/globalstats.php?year=2011>. – Загол. з екрану.
8. Браузер [Електронний ресурс] // Википедія. – Режим доступу : URL : <http://ru.wikipedia.org/wiki/Браузер>. – Загол. з екрану.
9. JavaScript framework statistics [Електронний ресурс]. – Режим доступу : URL : <http://wappalyzer.com/stats/cat/JavaScript%20frameworks>. – Загол. з екрану.
10. Рисунок Total known sites [Електронний ресурс]. – Режим доступу : URL : <http://moodle.org/stats/>. – Загол. з екрану.
11. Богомолов В. А. Обзор бесплатных систем управления обучением [Електронний ресурс] // Образовательные технологии и общество. – 2007. – Т. 10, № 3. – С. 439–460. – Режим доступу : URL : http://ifets.ieee.org/russian/depository/v10_i3/html/9_bogomolov.htm.
12. Боллобаш Н. М. Розробка дистанційного курсу засобами інформаційного середовища Moodle [Електронний ресурс] / Н. М. Боллобаш. – Режим доступу : URL : http://www.nbu.gov.ua/Portal/Soc_Gum/Npchdu/Pedagogics/2010_123/123-4.pdf. – Загол. з екрану.
13. Linux Distributions – Facts and Figures. (Tuesday 10 May 2011). How popular is your distribution? [Електронний ресурс]. – Режим доступу : URL : <http://distrowatch.com/stats.php?section=popularity>. – Загол. з екрану.
14. The Gaea Times. Top 5 Free Linux Distributions for Servers in 2010. [Електронний ресурс]. – Режим доступу : URL : <http://tech.gaeatimes.com/index.php/archive/top-5-free-linux-distributions-for-servers-in-2010/>. – Загол. з екрану.